MDCN: Multi-Scale, Deep Inception Convolutional Neural Networks for Efficient Object Detection

Wenchi Ma\textsuperscript{1}; Yuanwei Wu\textsuperscript{1}; Zongbo Wang\textsuperscript{2}; Guanghui Wang\textsuperscript{1}
\textsuperscript{1}University of Kansas, \textsuperscript{2}Ainstein Inc.

Motivation
Object detection in challenging situations such as scale variation, occlusion, and truncation depends not only on feature details but also on contextual information.

- Previous: emphasize much on detail features by deeper and wider network
- Problem: low effectiveness of feature usage with high load of computation as feature details are easily being changed or even “washed out” after passing through complicated filtering structures.
- MDCN: proposes multi-scale and deep inception convolutional neural network, focusing on wider and broader object regions by activating feature maps produced in deep part of the network.

Detection Pipeline
Feature extraction, wide-angle contextual information, object classification and bounding box regression are performed in a single-shot pipeline.

1. Base network: VGG-16
   - extract high-resolution, low-dimensional features
2. Multi-scale deep inception module:
   - extract object main-body and multi-scale contextual information.

Information-Square Inception Modules
- Combination of 1x1, 3x3 and 5x5 filters: activating multi-scale receptive fields
- using two series of 3x3 filters to replace 5x5 filter so as to minimize the number of parameters

By defining weights to each filtering units, the information-square inception modules formed.

\[ F = f_j(f_1(\Phi_j)) + 2 \times f_2(\Phi_j) + \Phi_{j+1}, m \leq k \leq m \]

\[ F_j(\Phi_j) = (f_j + 1)(\Phi_j), m \leq k \leq m \]

Data and Implementation
Dataset: KITTI
- containing many challenging objects like small and occluded cars, pedestrians and cyclists
- objects are labeled as easy, moderate, and hard based on how much objects are occluded and truncated
- All images are rescaled from 1242x375 to 300x300
- Intersection over Union (IoU) for car, pedestrian and cyclist are all set to 50%
- The VGG-16 base network is pretrained on ImageNet and MDCN is fine-tuned on KITTI

Detection Accuracy

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|c|c|c|c|c|c|}
\hline
\textbf{Model} & \textbf{Car} & \textbf{Pedestrian} & \textbf{Car} & \textbf{Pedestrian} & \textbf{Car} & \textbf{Pedestrian} & \textbf{Car} & \textbf{Pedestrian} & \textbf{IoU} & \textbf{mAP} \\
\hline
\hline
\textbf{RoVA} & 87.37 & 75.00 & 88.00 & 75.00 & 87.9 & 75.00 & 87.9 & 75.00 & 75.00 & 75.00 \\
\textbf{VGG-16} & 86.00 & 75.00 & 87.9 & 75.00 & 87.9 & 75.00 & 87.9 & 75.00 & 75.00 & 75.00 \\
\textbf{Kinetik} & 65.00 & 65.00 & 65.00 & 65.00 & 65.00 & 65.00 & 65.00 & 65.00 & 65.00 & 65.00 \\
\textbf{MS Inception} & 50.00 & 50.00 & 50.00 & 50.00 & 50.00 & 50.00 & 50.00 & 50.00 & 50.00 & 50.00 \\
\textbf{MDCN} & 86.76 & 86.76 & 86.76 & 86.76 & 86.76 & 86.76 & 86.76 & 86.76 & 86.76 & 86.76 \\
\hline
\end{tabular}
\end{table}

Fig. 1. Multi-scale, wide-context receptive field activation

Fig. 2. The architecture of MDCN. The wide-context, multi-scale deep inception module consists of multiple filtering structures. The red, yellow and green boxes each indicate one filter size.

TABLE II

<table>
<thead>
<tr>
<th>Class</th>
<th>Model</th>
<th>Width</th>
<th>Height</th>
</tr>
</thead>
<tbody>
<tr>
<td>Car</td>
<td>VGG-16</td>
<td>600</td>
<td>400</td>
</tr>
<tr>
<td>Pedestrian</td>
<td>VGG-16</td>
<td>600</td>
<td>400</td>
</tr>
<tr>
<td>Cyclist</td>
<td>VGG-16</td>
<td>600</td>
<td>400</td>
</tr>
</tbody>
</table>

TABLE III

<table>
<thead>
<tr>
<th>Model</th>
<th>Network</th>
<th>LPC</th>
<th>Resolution</th>
<th>F of Average</th>
<th>FPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>MDCN1 &amp; VGG-16</td>
<td>4.0 &amp; 500x500</td>
<td>2.0 &amp; 300 &amp; 12.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MDCN2 &amp; VGG-16</td>
<td>4.0 &amp; 500x500</td>
<td>2.0 &amp; 300 &amp; 12.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MDCN3 &amp; VGG-16</td>
<td>4.0 &amp; 500x500</td>
<td>2.0 &amp; 300 &amp; 12.0</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>